DD-IDDE: GOD MODE FOR DESIGNING SCENARIO-DRIVEN SKILLS IN DEEPPAVLOV DREAM

Daniel Kornev

CPO @ DeepPavlov

alexa prize

Grand Challenge: create a socialbot that can engage in a fun, high quality conversation on popular societal topics for 20 minutes and achieve an average rating of at least 4.0/5.0.

Alexa Prize 3 vs 4 Winners:

1. $500K, 3.8 vs 3.28, 7 min 32 sec vs 14 min 14 sec
2. $100K 3.17 vs 3.25

Alexa Prize 4 Winner:

Alquist (2nd in ‘17, 3rd in `18,`20) – Czech Technical University

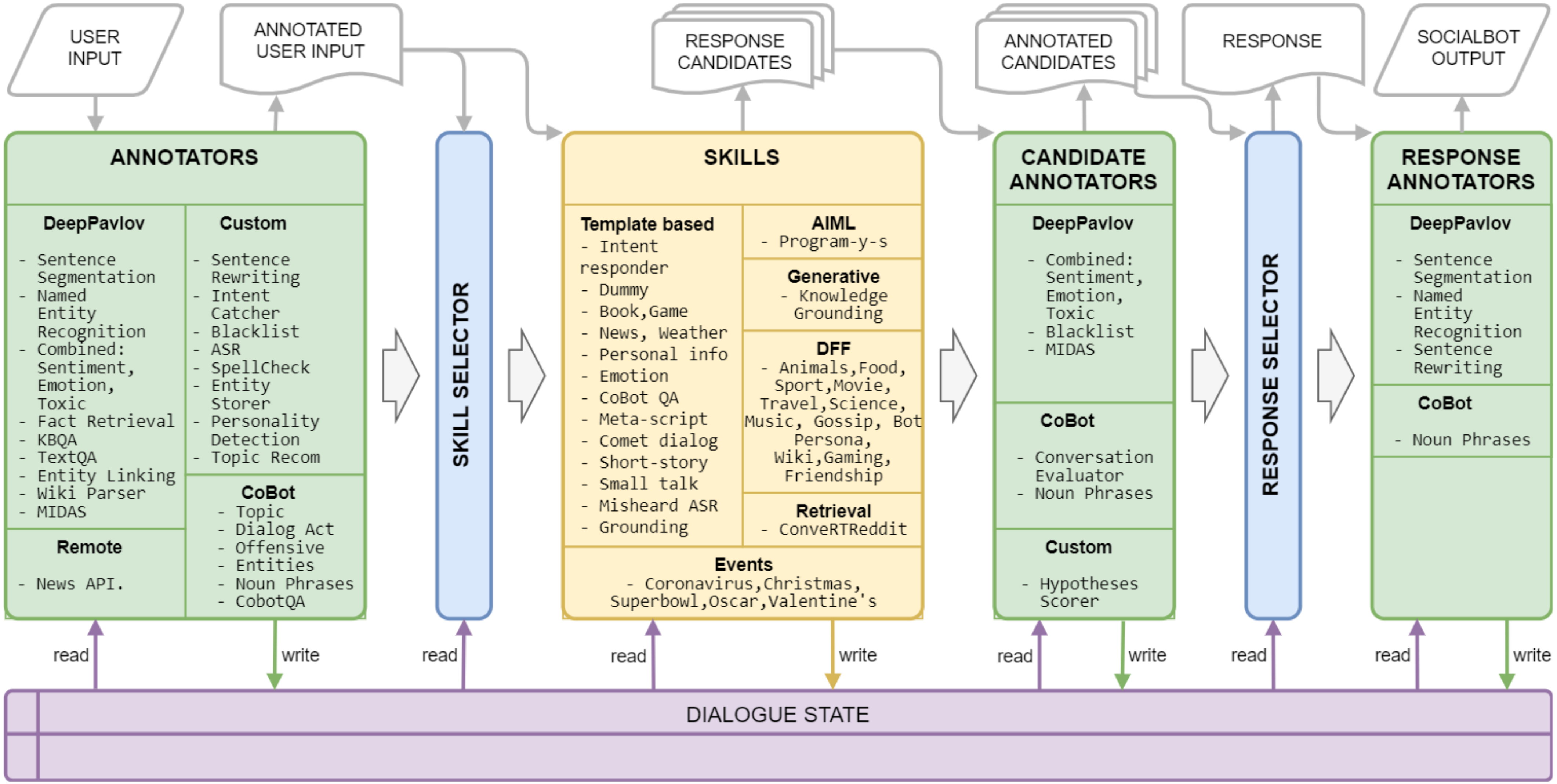
WHY NEUROSYMBOLIC?

"The aim is to identify a way of looking at and manipulating commonsense knowledge that is consistent with and can support what we consider to be the two most fundamental aspects of intelligent cognitive behaviour: the ability to learn from experience and the ability to reason from what has been learned. We are therefore seeking a semantics of knowledge that can computationally support the basic phenomena of intelligent behaviour."

Leslie Valiant

DREAM FOR alexa prize

DREAM SOCIALBOT: ALEXA PRIZE 4



FINITE STATE MACHINE FX...

Python–Native

pip install

Lots of extensions

Battle-tested

Extensions

Entity Detection

Intent Classifier

Dialog Acts & Speech Functions

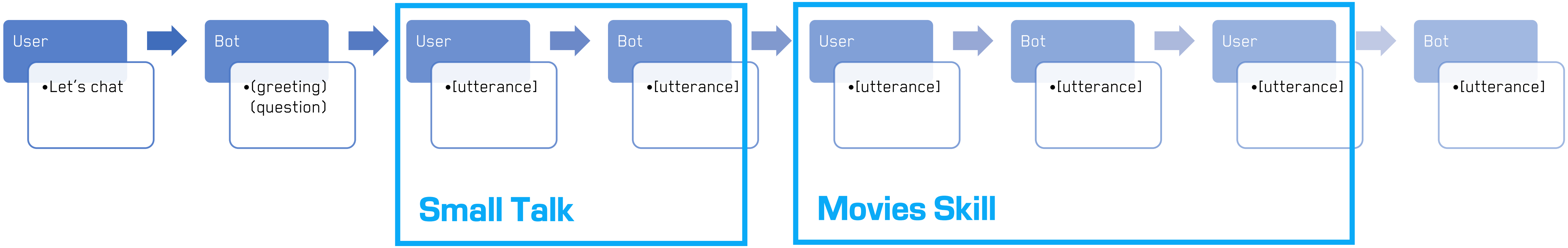
Dialog Analysis Dashboard

examples > animals.py > ...

```python
import logging

from dff.core.keywords import TRANSITIONS, RESPONSE, MISC
from dff.core import Actor
import dff.conditions as cnd

from utils import condition as loc_cnd
from utils import common


logger = logging.getLogger(__name__)

plot = {
    "greeting_flow": {
        "start_node": {
            TRANSITIONS: {"greet_and_ask_about_pets": cnd.all([loc_cnd.is_sf("Op
            RESPONSE: "Hi %username%!",
            MISC: {"speech_functions": ["Open.Attend"]},
        },
        "fallback_node": {TRANSITIONS: {}, RESPONSE: "Ooops"},
        "greet_and_ask_about_pets": {
            TRANSITIONS: {
                "cool_and_clarify_which_pets": cnd.any(
                    [
                        cnd.any(
                            [
                                loc_cnd.is_sf("React.Respond.Support.Reply.Agree
                                loc_cnd.is_sf("React.Respond.Support.Reply.Affir
                            ]
                        ),
                        cnd.all([loc_cnd.is_sf("React.Rejoinder"), loc_cnd.is_mi
                    ]
                ),
                "sad_and_say_about_pets": cnd.any(
```

Finite State Machine – Scenario-Driven Approach

Scenario-Driven approach enables sequential experience and is strategic, yet it takes a long time and still misses on what user can say.



Pros: Scenario-Driven skills enabled us to give users **several conversation turns with the same context**, and limited user understanding within human-curated scenarios

Cons: given that we can't predict what user might say, lots of iterations needed.

DESIGNING OPEN-DOMAIN CHATBOTS

Designing Open-Domain Chatbots is a complex process.

Problem

User can say anything

Bot: Hi, how are you doing?

User: Hi! User: What's up? User: I'm sad. User: Had an awful convo. User: Switch off lights

Interpretation

How to recognize pattern?

Regular Expressions: Require detailed scenarios and long iterative development process

Intents: Work only within the given domain (task-oriented chatbots) or at high-level (greeting, etc.)

Dialog Acts: Work in limited cases (yes/no, question/statement)

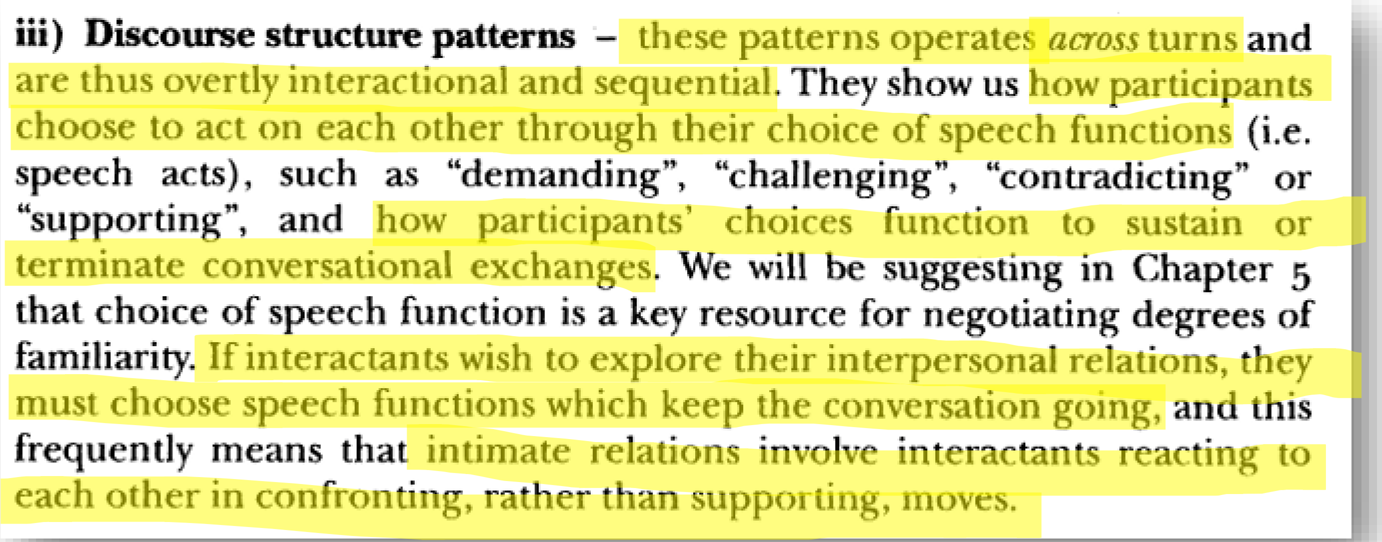
Response

How to react?

WELCOME DISCOURSE

Discourse structure patterns operate *across* turns: thus overtly interactional & sequential

Eggins and Martin (1997)



iii) **Discourse structure patterns** – these patterns operates *across* turns and are thus overtly interactional and sequential. They show us how participants choose to act on each other through their choice of speech functions (i.e. speech acts), such as "demanding", "challenging", "contradicting" or "supporting", and how participants' choices function to sustain or terminate conversational exchanges. We will be suggesting in Chapter 5 that choice of speech function is a key resource for negotiating degrees of familiarity. If interactants wish to explore their interpersonal relations, they must choose speech functions which keep the conversation going, and this frequently means that intimate relations involve interactants reacting to each other in confronting, rather than supporting, moves.

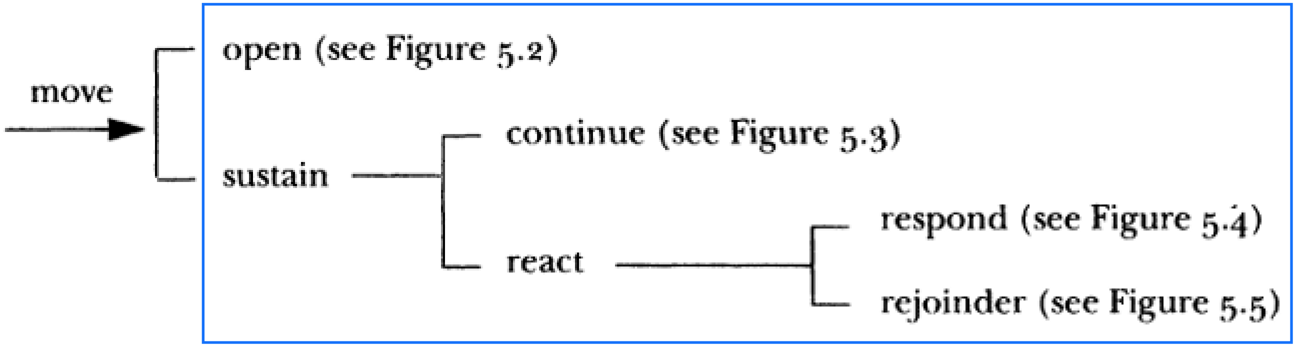
Discourse Management is a basis for acting *across* turns, thus becoming strategic

HOW DOES DISCOURSE MANAGEMENT WORK?

Speech Functions control Discourse:

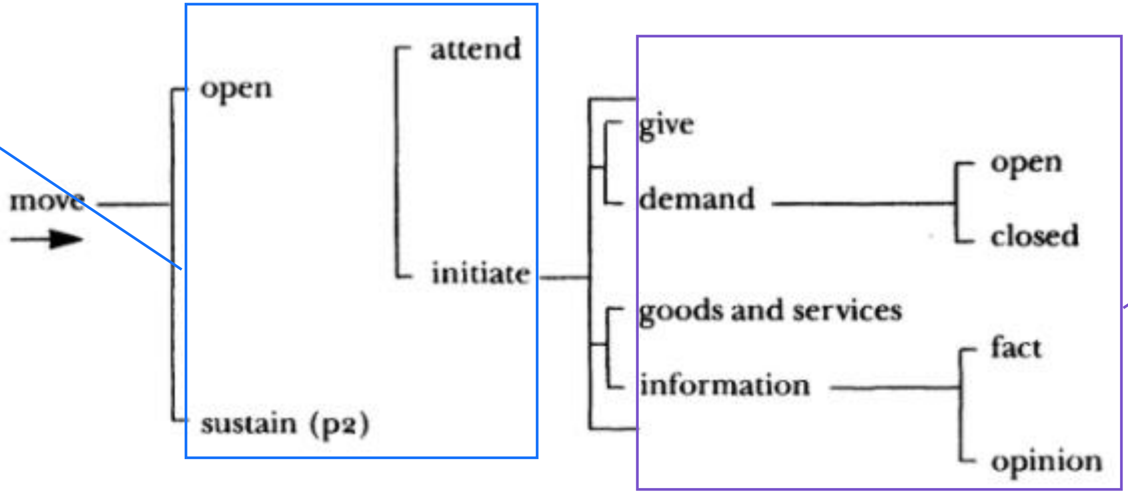
Eggins and Slade (1997)

Discourse Moves



Speech Acts

Give information
Demand information



Speech Function Example:
`open:initiate:give_opinion`

DD-IDDE

IDQQD

God Mode for

For Dialogue Designers

for Doom Players

DD-IDDE

Open–Source Dev Tool

Extension for VS Code

Draw.io-based

Draw.io <-> Python DSL of DFF

Aids in Rapid Dialog Design

With Discourse Recommendations…

Statistics…

Plugins!

DD-IDDE v0.2.28

DeepPavlov.ai | 57 | ☆☆☆☆☆

Discourse-Driven Integrated Dialog Development Environment.

Disable | Uninstall

Extension is enabled on 'WSL: Ubuntu-18.04'

Details | Feature Contributions | Changelog | Runtime Status

DD-IDDE: Design Discourse-Driven Open-Domain Chatbots

Follow @deeppavlov 664

This extension enables developers of the open-domain chatbots with an option to design them within VS Code using customized Draw.io UI and **Discourse Moves Recommendations** to make dialog more smooth and natural.

This extension uses DeepPavlov's Dialog Flow Framework as the runtime environment for the open-domain chatbots. It also uses DD-IDDE SDK that enables the use of the Discourse Moves recommendations during the design phase as well in the runtime.

Discourse Moves Recommendation System has been built based on the Discourse & Speech Functions theory originally created by M.A.K. Halliday and further developed by Eggins & Slade in their book "Analysing Casual Conversation".

This extension is built on top of the unnoficial integration of the Draw.io (also known as diagrams.net) into VS Code made by Henning Dieterichs, hediet on Github (Main Contributor / Author).

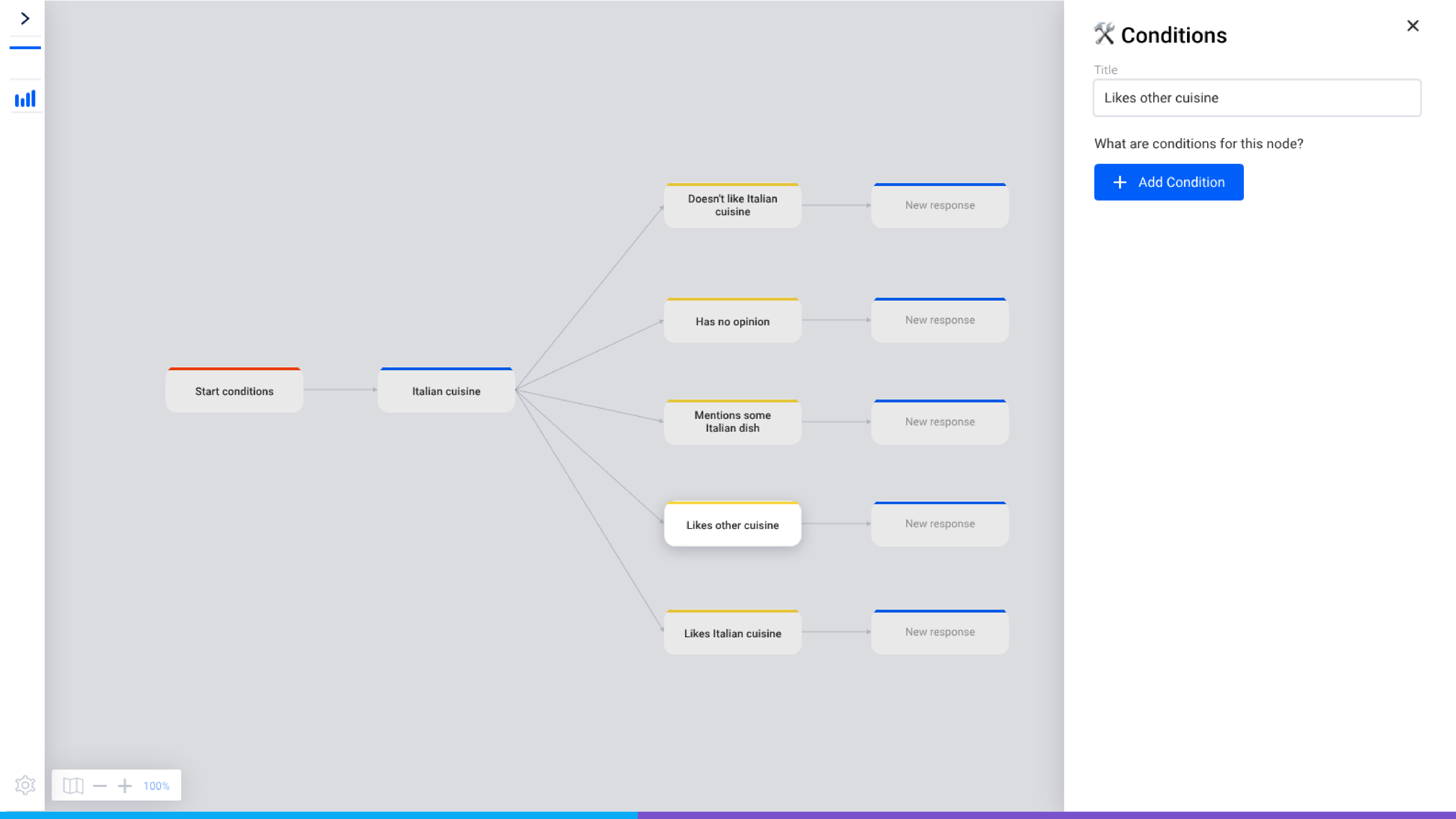
Features

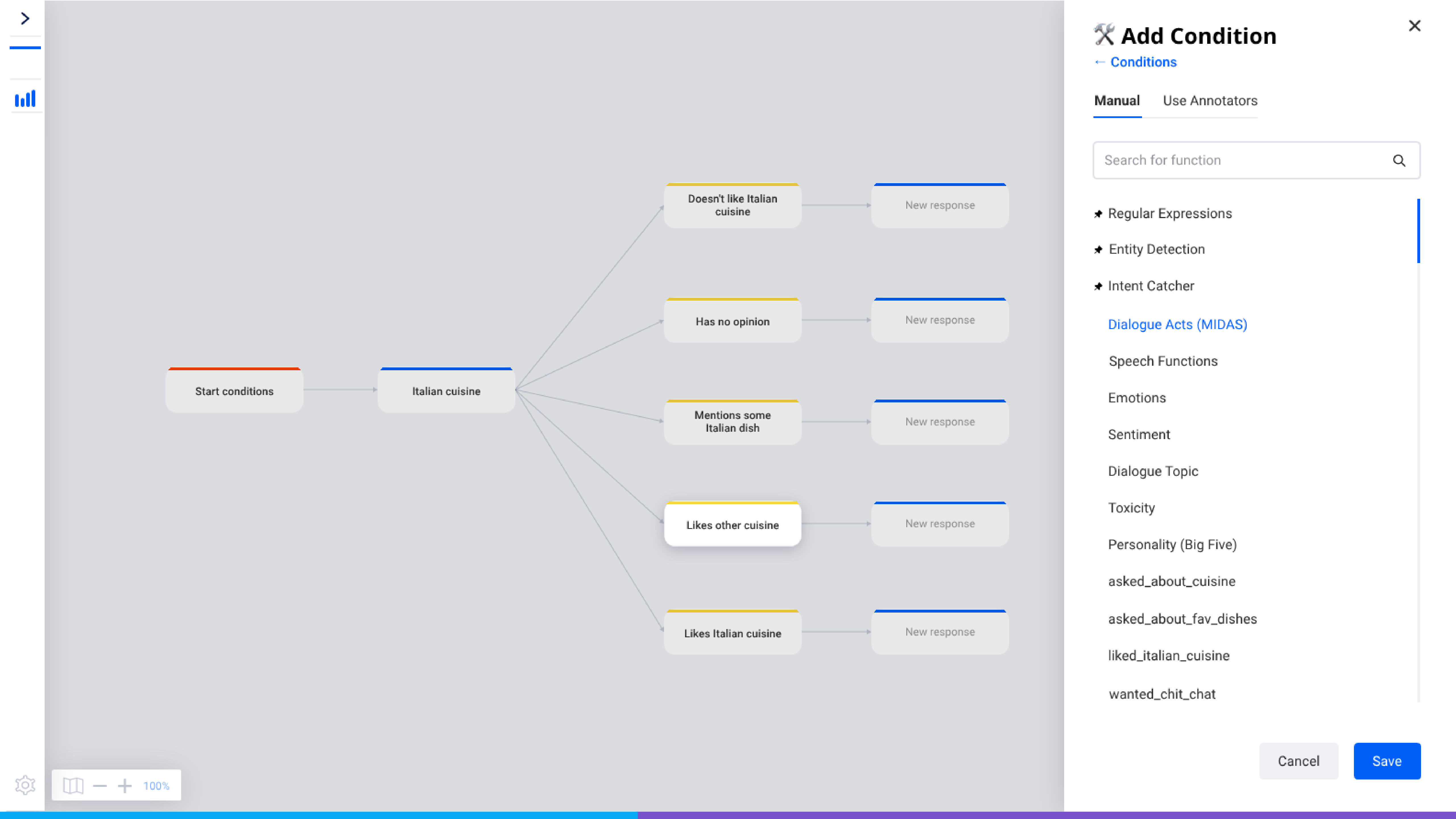
- Open `.py` files representing dialog flows of the open-domain chatbots using the Draw.io editor.

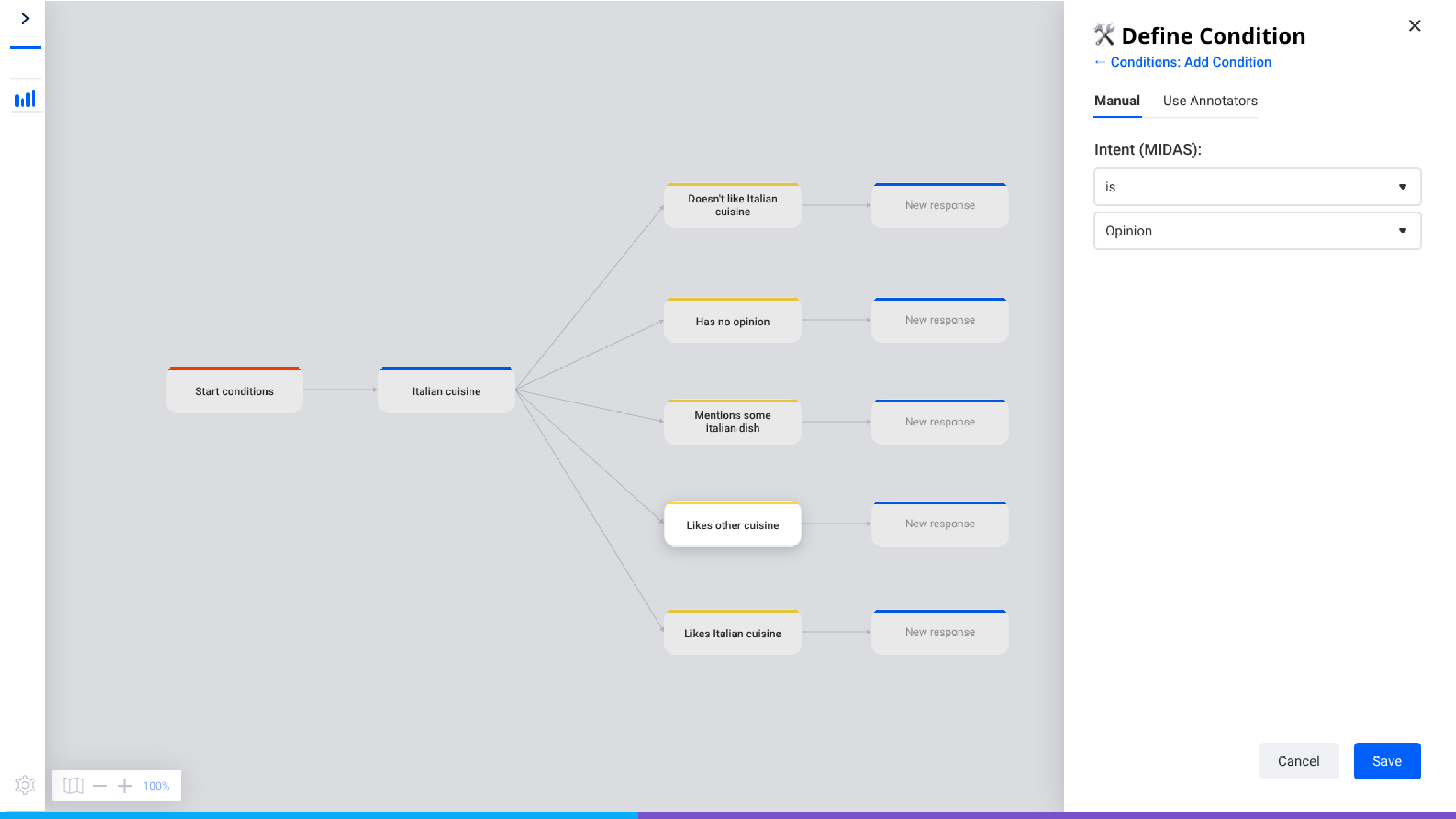
DEMO

DD-IDDE: WHAT'S NEXT?

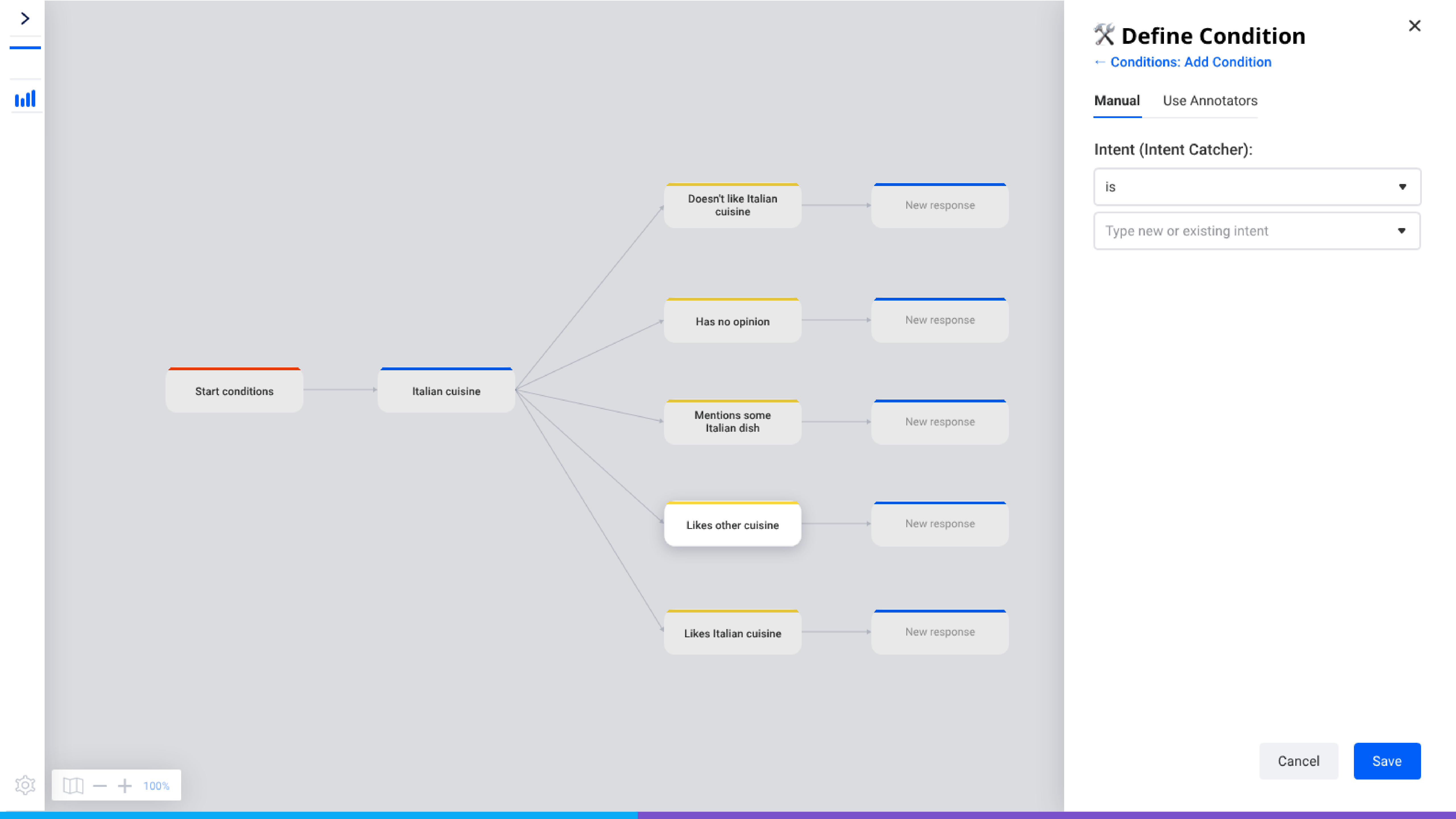
Custom spatial UI, re-built suggestions, auto response generation and… a new name!

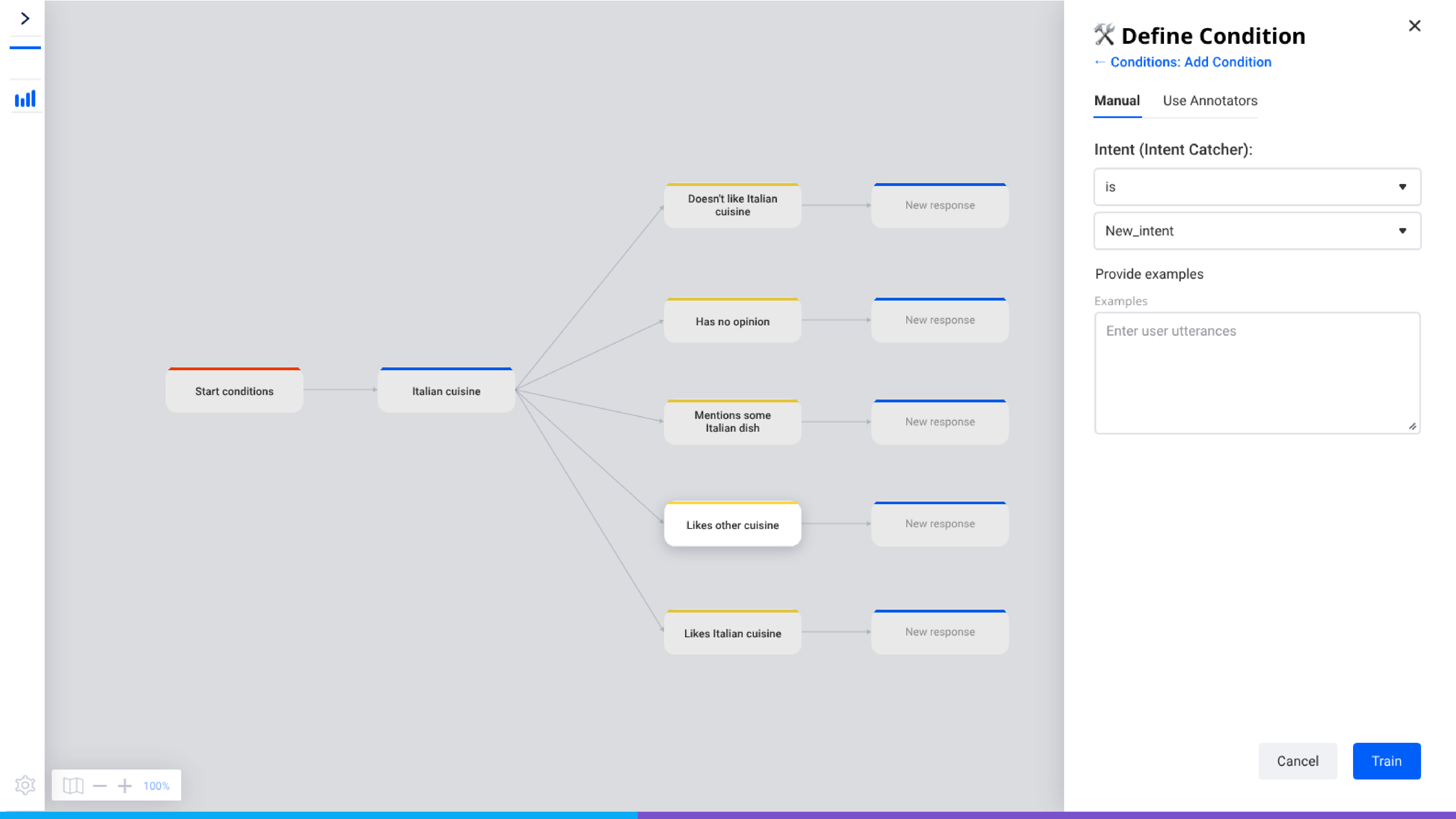

Conditions
Title
Likes other cuisine
What are conditions for this node?
Add Condition
Start conditions
Italian cuisine
Doesn't like Italian cuisine
Has no opinion
Mentions some Italian dish
Likes other cuisine
Likes Italian cuisine
New response
New response
New response
New response
New response
100%

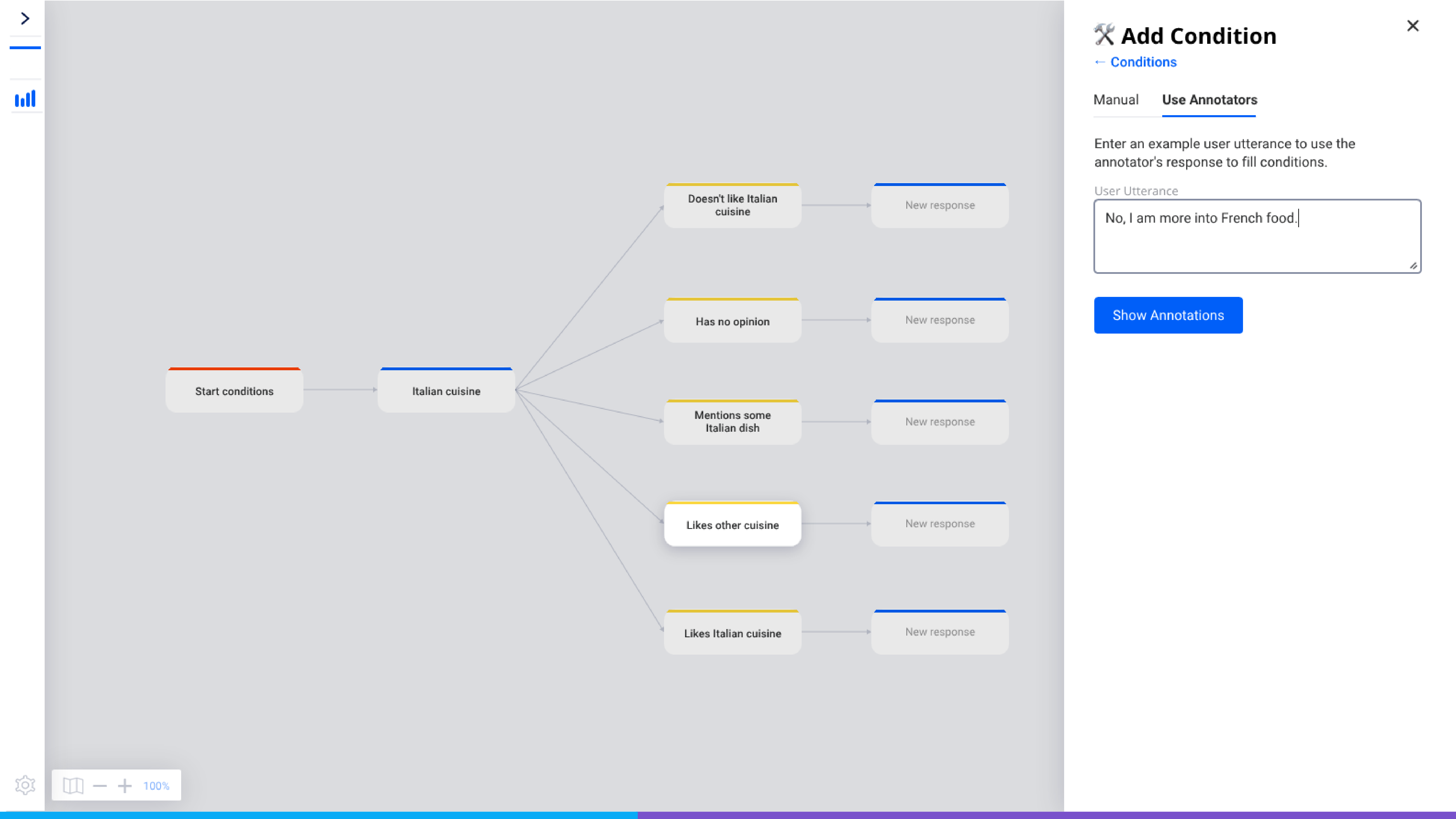

Start conditions
Italian cuisine
Doesn't like Italian cuisine
Has no opinion
Mentions some Italian dish
Likes other cuisine
Likes Italian cuisine
New response
New response
New response
New response
New response
100%
Add Condition
← Conditions
Manual
Use Annotators
Search for function
Regular Expressions
Entity Detection
Intent Catcher
Dialogue Acts (MIDAS)
Speech Functions
Emotions
Sentiment
Dialogue Topic
Toxicity
Personality (Big Five)
asked_about_cuisine
asked_about_fav_dishes
liked_italian_cuisine
wanted_chit_chat
Cancel
Save

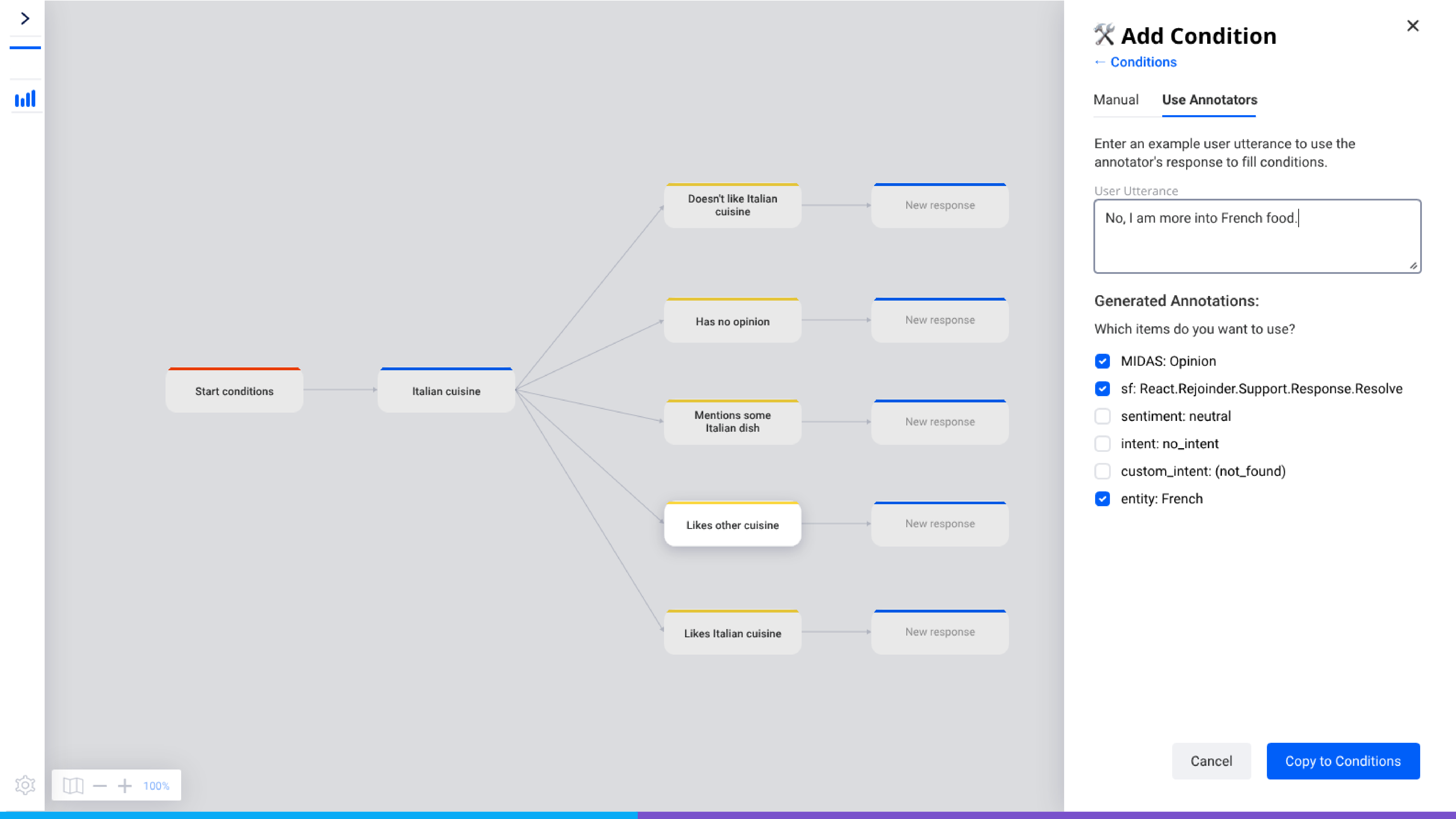


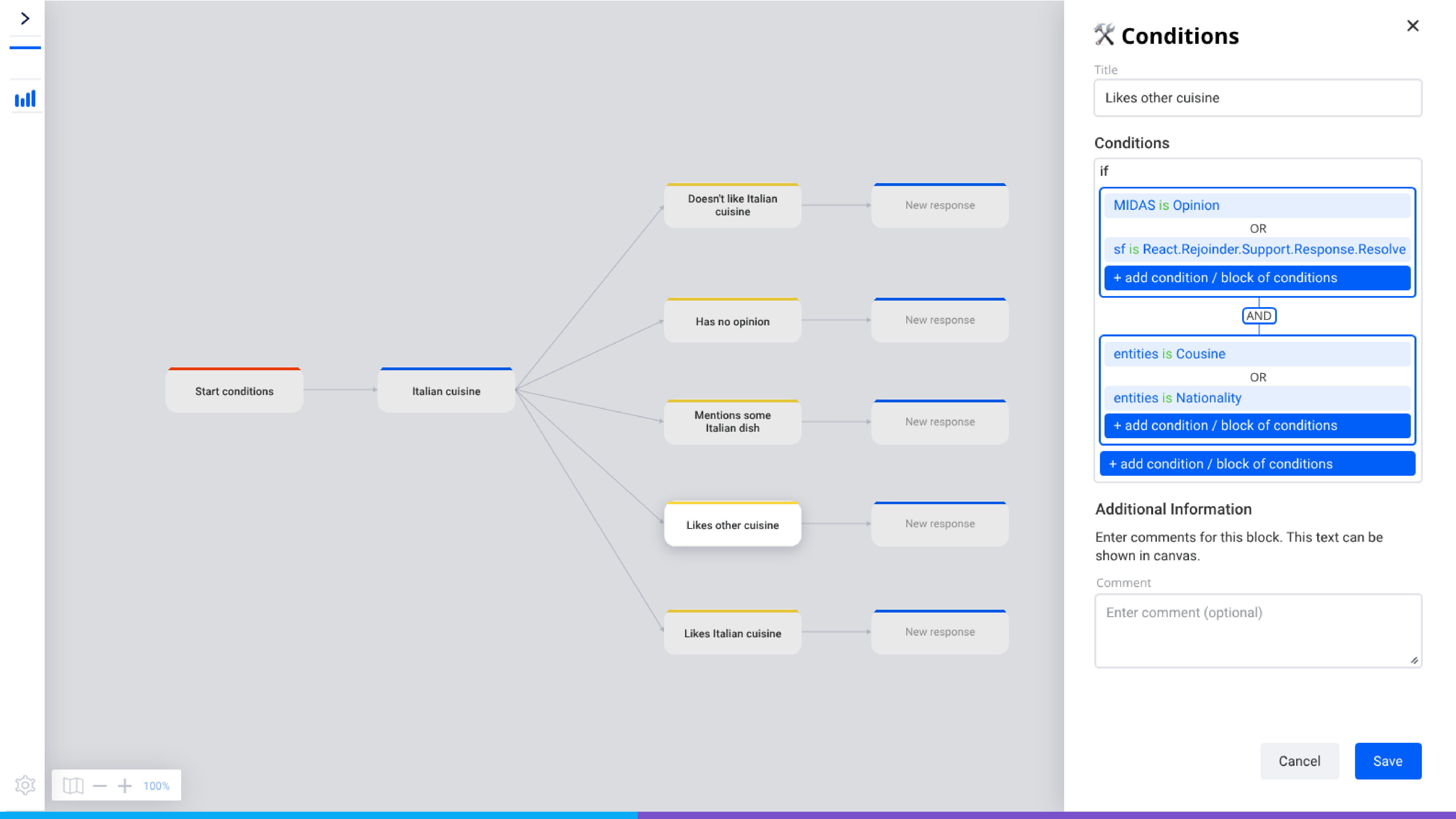
Start conditions
Italian cuisine
Doesn't like Italian cuisine
Has no opinion
Mentions some Italian dish
Likes other cuisine
Likes Italian cuisine
New response
New response
New response
New response
New response
100%
Define Condition
← Conditions: Add Condition
Manual
Use Annotators
Intent (MIDAS):
is
Opinion
Cancel
Save

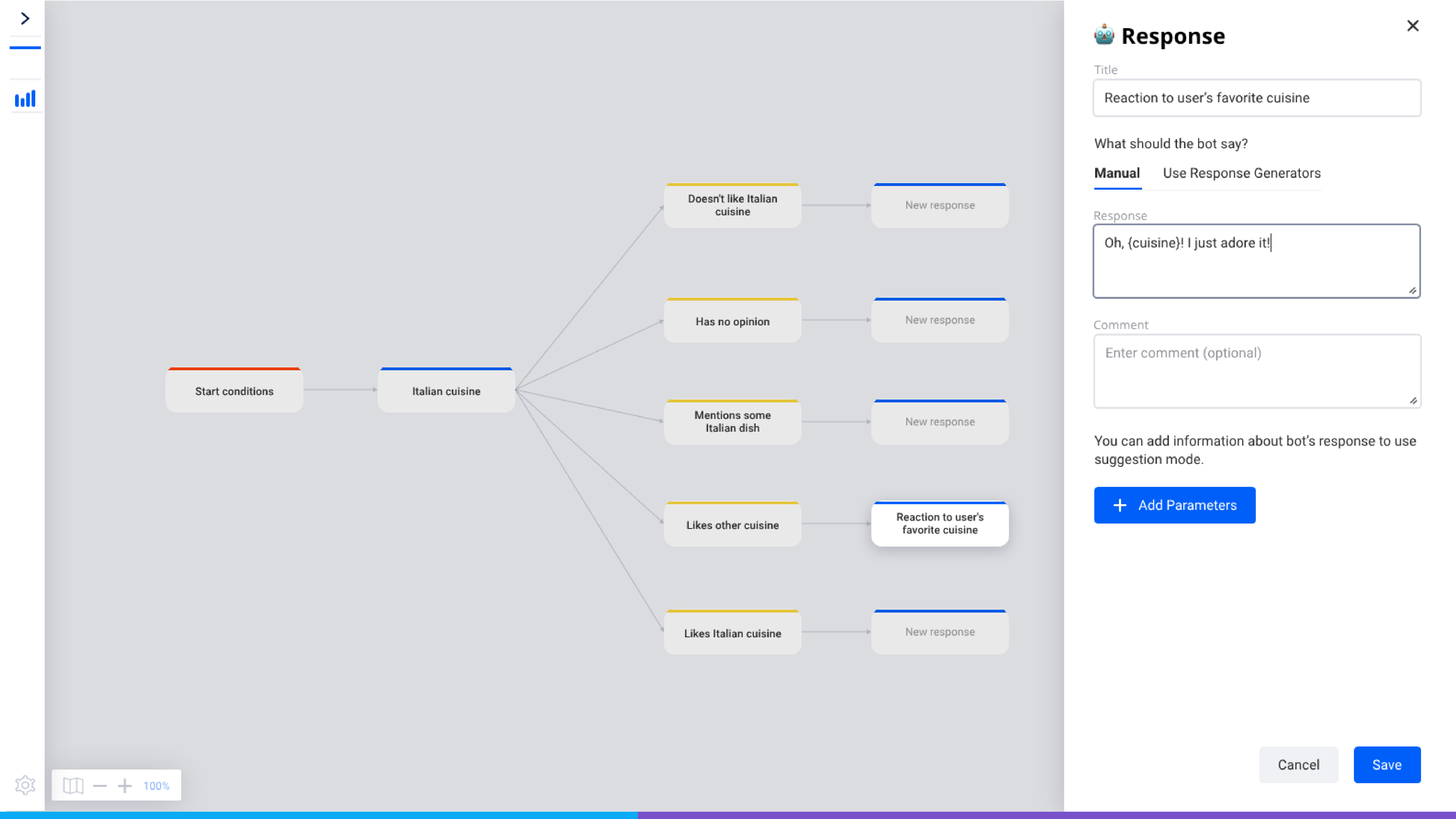

Start conditions
Italian cuisine
Doesn't like Italian cuisine
Has no opinion
Mentions some Italian dish
Likes other cuisine
Likes Italian cuisine
New response
New response
New response
New response
New response
100%
Define Condition
← Conditions: Add Condition
Manual
Use Annotators
Intent (Intent Catcher):
is
Type new or existing intent
Cancel
Save

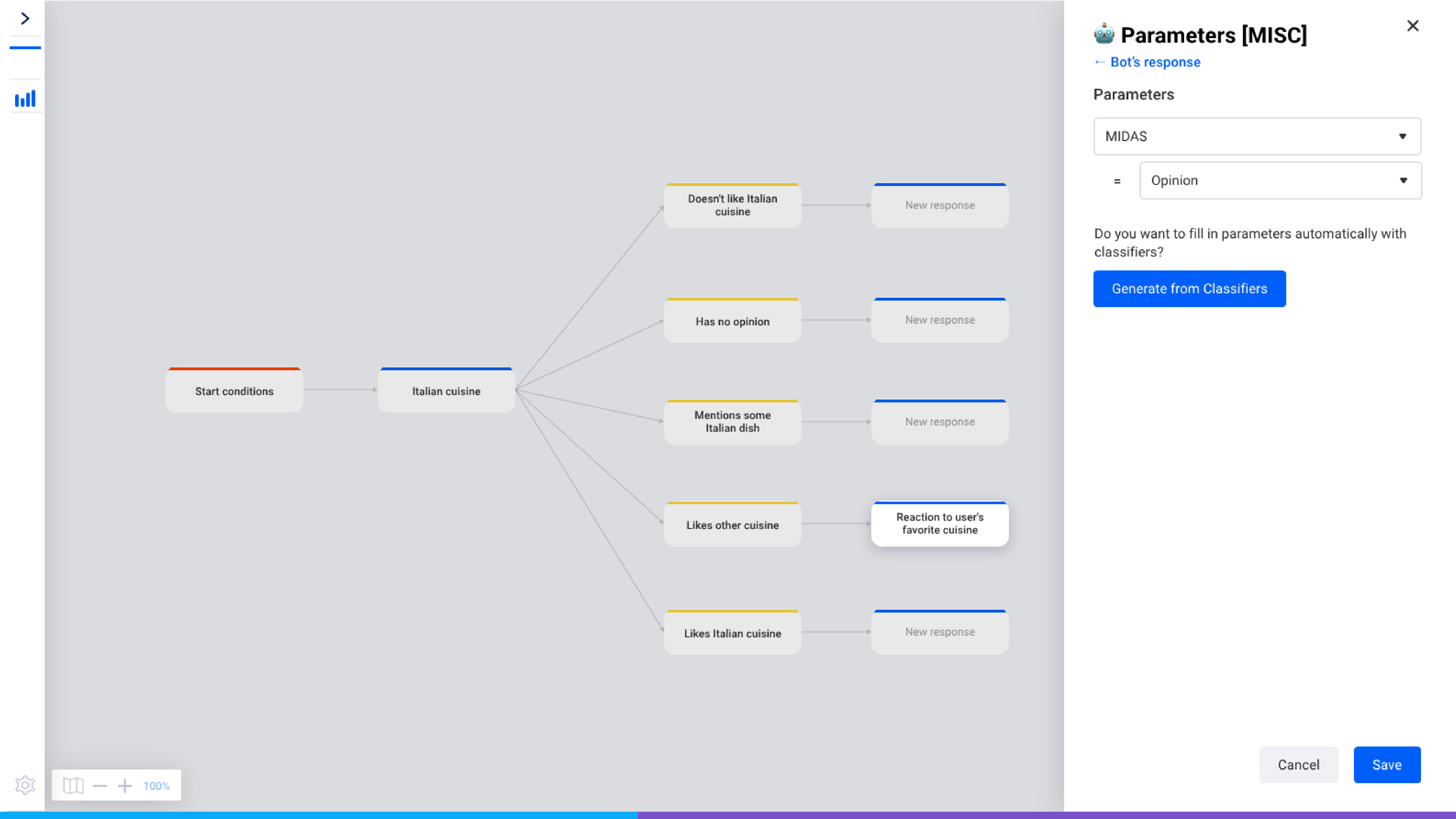

Start conditions
Italian cuisine
Doesn't like Italian cuisine
Has no opinion
Mentions some Italian dish
Likes other cuisine
Likes Italian cuisine
New response
New response
New response
New response
New response
100%
Define Condition
← Conditions: Add Condition
Manual
Use Annotators
Intent (Intent Catcher):
is
New_intent
Provide examples
Examples
Enter user utterances
Cancel
Train

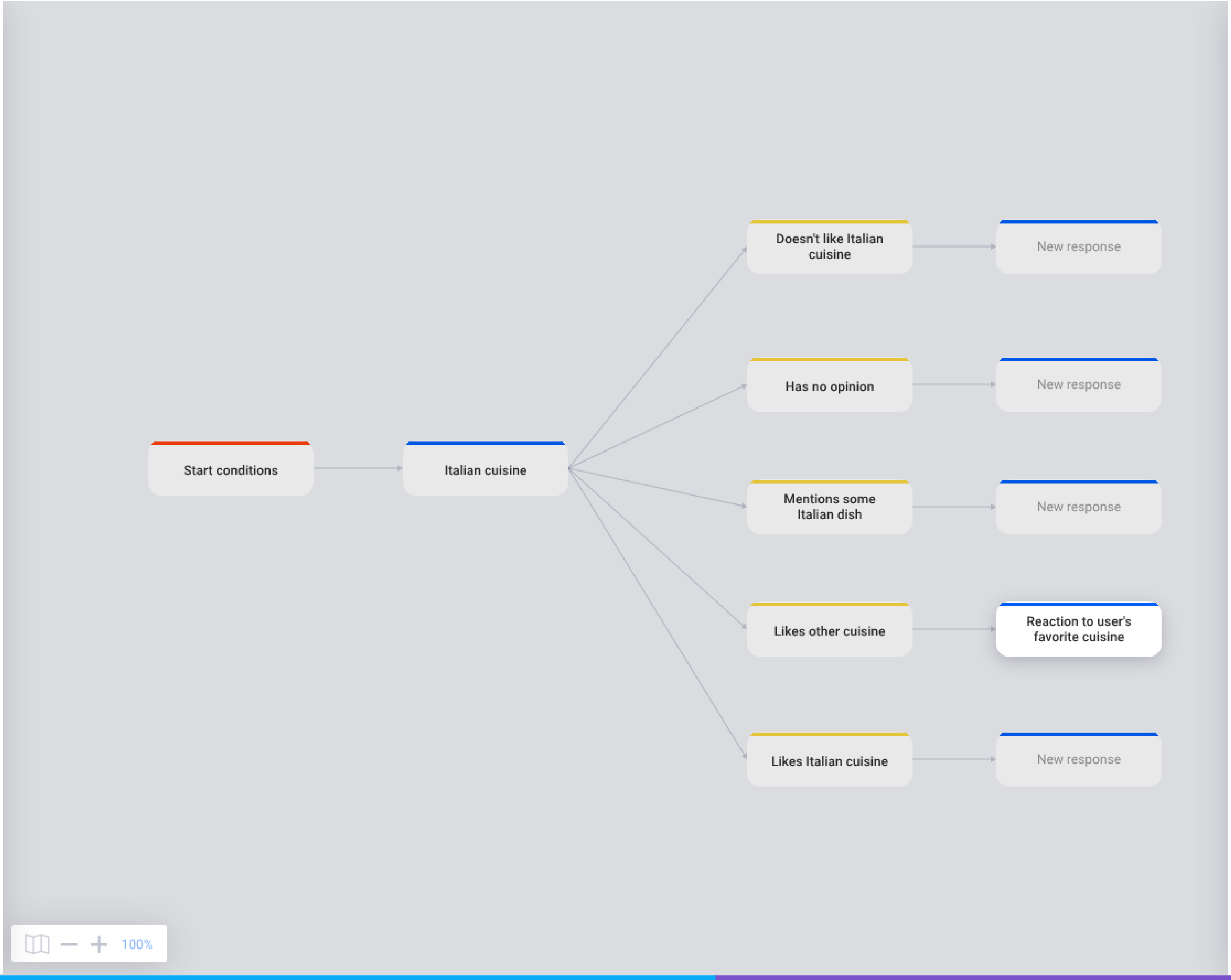

Start conditions
Italian cuisine
Doesn't like Italian cuisine
Has no opinion
Mentions some Italian dish
Likes other cuisine
Likes Italian cuisine
New response
New response
New response
New response
New response
100%
Add Condition
← Conditions
Manual
Use Annotators
Enter an example user utterance to use the annotator's response to fill conditions.
User Utterance
No, I am more into French food.
Show Annotations


Add Condition
← Conditions
Manual
Use Annotators
Enter an example user utterance to use the annotator's response to fill conditions.
User Utterance
No, I am more into French food.
Generated Annotations:
Which items do you want to use?
MIDAS: Opinion
sf: React.Rejoinder.Support.Response.Resolve
sentiment: neutral
intent: no_intent
custom_intent: (not_found)
entity: French
Cancel
Copy to Conditions
Start conditions
Italian cuisine
Doesn't like Italian cuisine
Has no opinion
Mentions some Italian dish
Likes other cuisine
Likes Italian cuisine
New response
New response
New response
New response
New response
100%


Start conditions
Italian cuisine
Doesn't like Italian cuisine
Has no opinion
Mentions some Italian dish
Likes other cuisine
Likes Italian cuisine
New response
New response
New response
New response
New response
100%
Conditions
Title
Likes other cuisine
Conditions
if
MIDAS is Opinion
OR
sf is React.Rejoinder.Support.Response.Resolve
+ add condition / block of conditions
AND
entities is Cousine
OR
entities is Nationality
+ add condition / block of conditions
+ add condition / block of conditions
Additional Information
Enter comments for this block. This text can be shown in canvas.
Comment
Enter comment (optional)
Cancel
Save


Response
Title
Reaction to user's favorite cuisine
What should the bot say?
Manual
Use Response Generators
Response
Oh, {cuisine}! I just adore it!
Comment
Enter comment (optional)
You can add information about bot's response to use suggestion mode.
Add Parameters
Cancel
Save
Start conditions
Italian cuisine
Doesn't like Italian cuisine
Has no opinion
Mentions some Italian dish
Likes other cuisine
Likes Italian cuisine
New response
New response
New response
Reaction to user's favorite cuisine
New response
100%


Start conditions
Italian cuisine
Doesn't like Italian cuisine
Has no opinion
Mentions some Italian dish
Likes other cuisine
Likes Italian cuisine
New response
New response
New response
Reaction to user's favorite cuisine
New response
100%
Parameters [MISC]
← Bot's response
Parameters
MIDAS
=
Opinion
Do you want to fill in parameters automatically with classifiers?
Generate from Classifiers
Cancel
Save



Parameters [MISC]

← Bot's response

Parameters

MIDAS

= Opinion

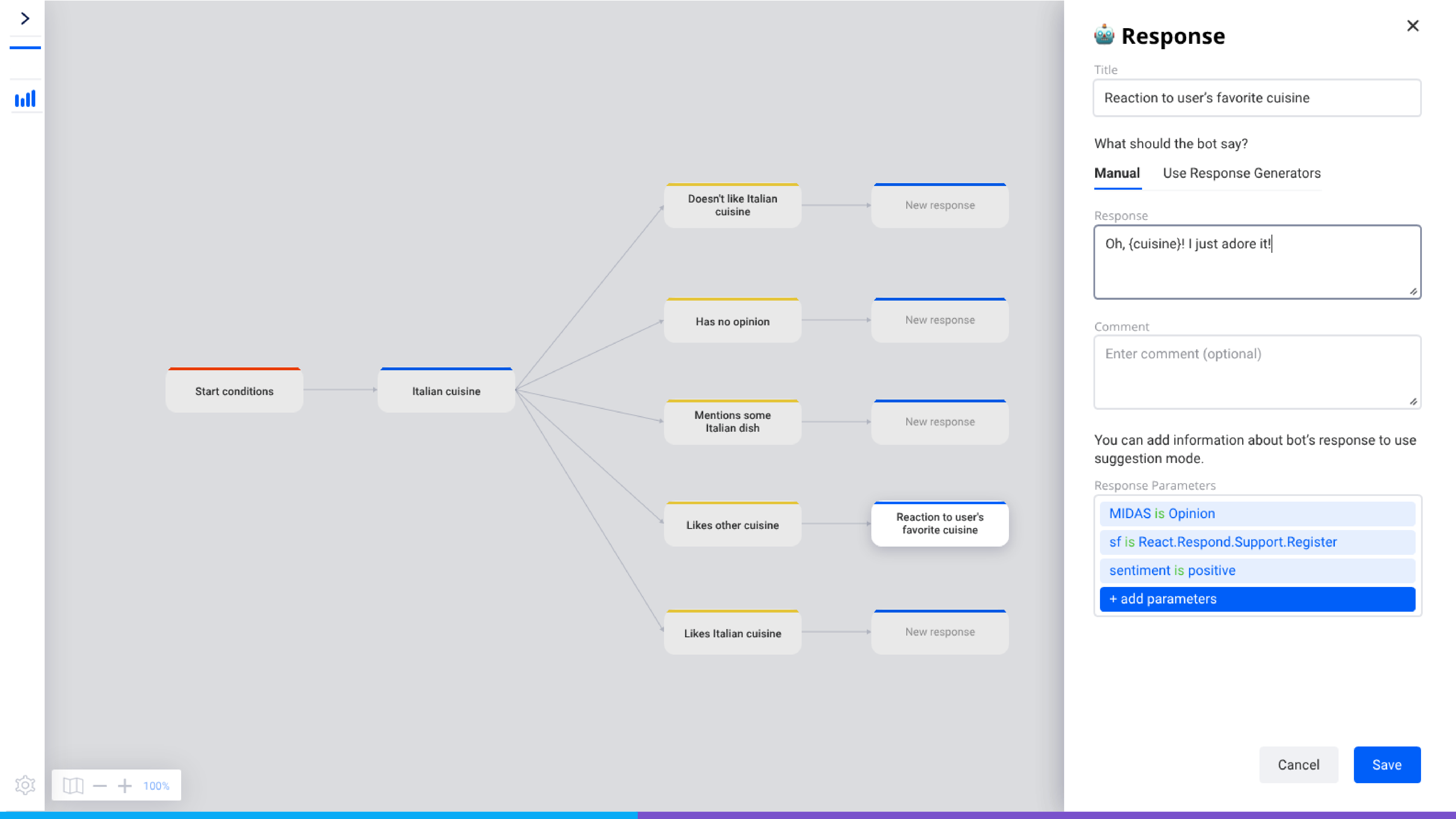
(Key)

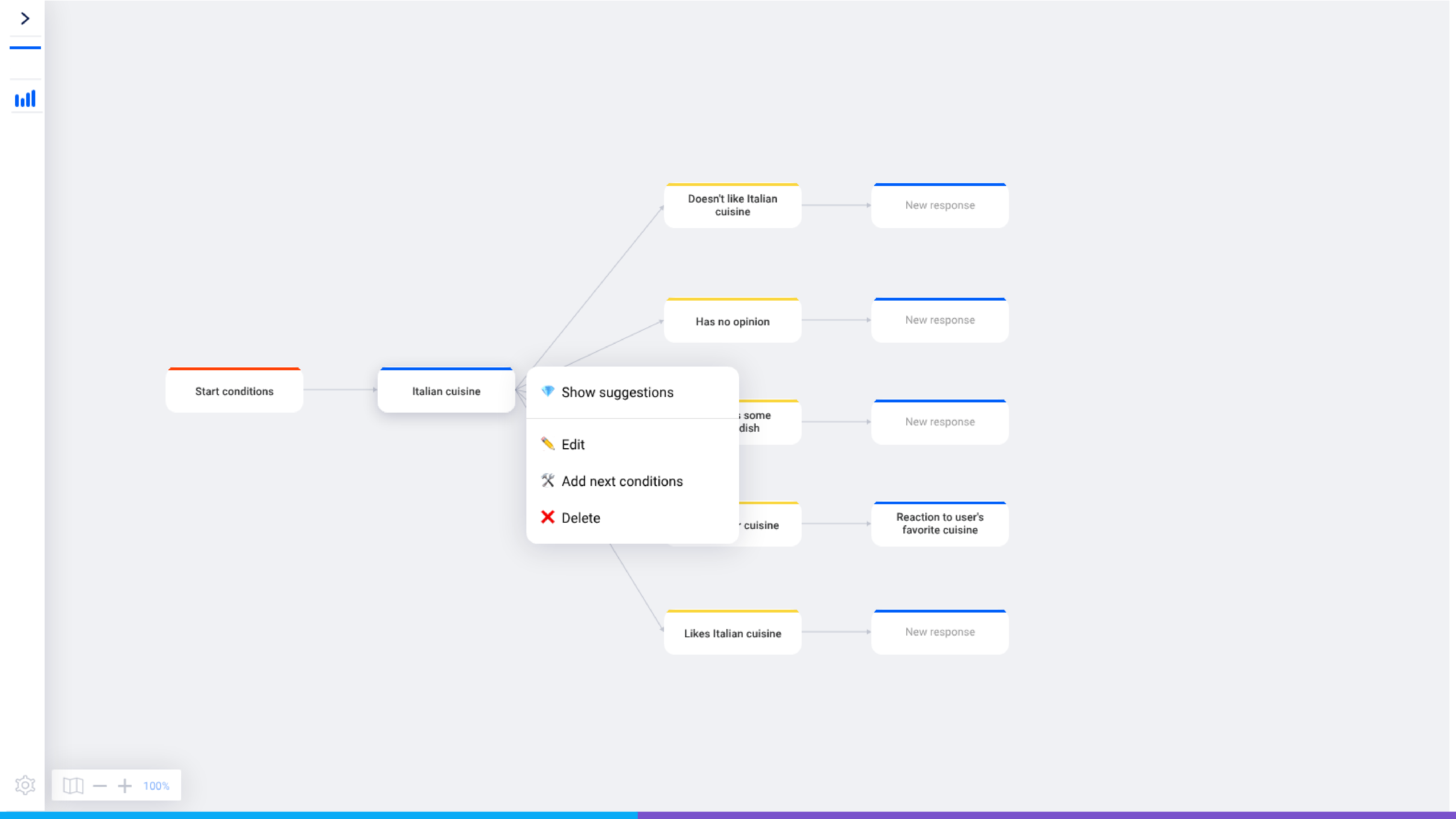
= (Value)

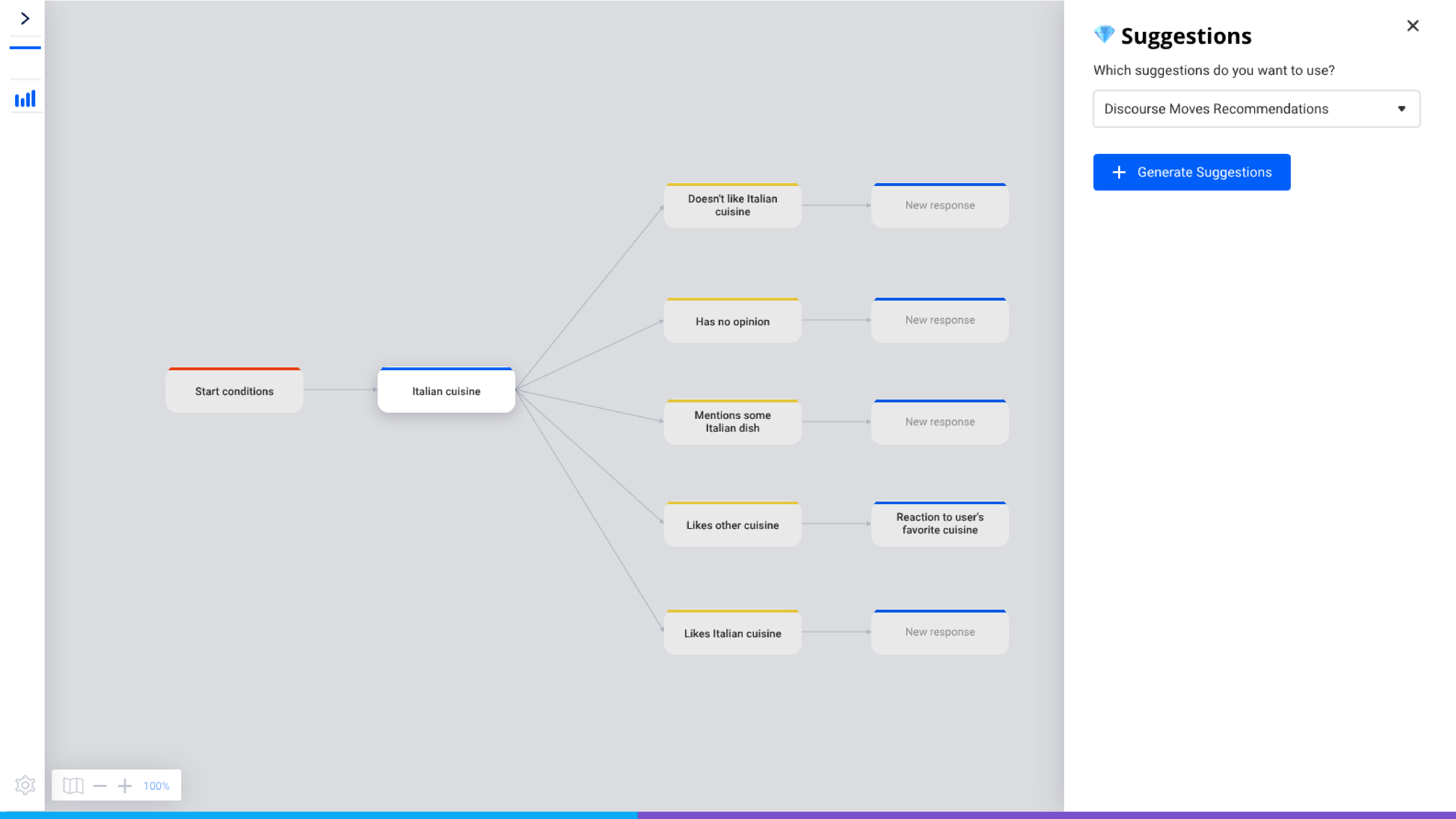
Do you want to fill in parameters automatically with classifiers?

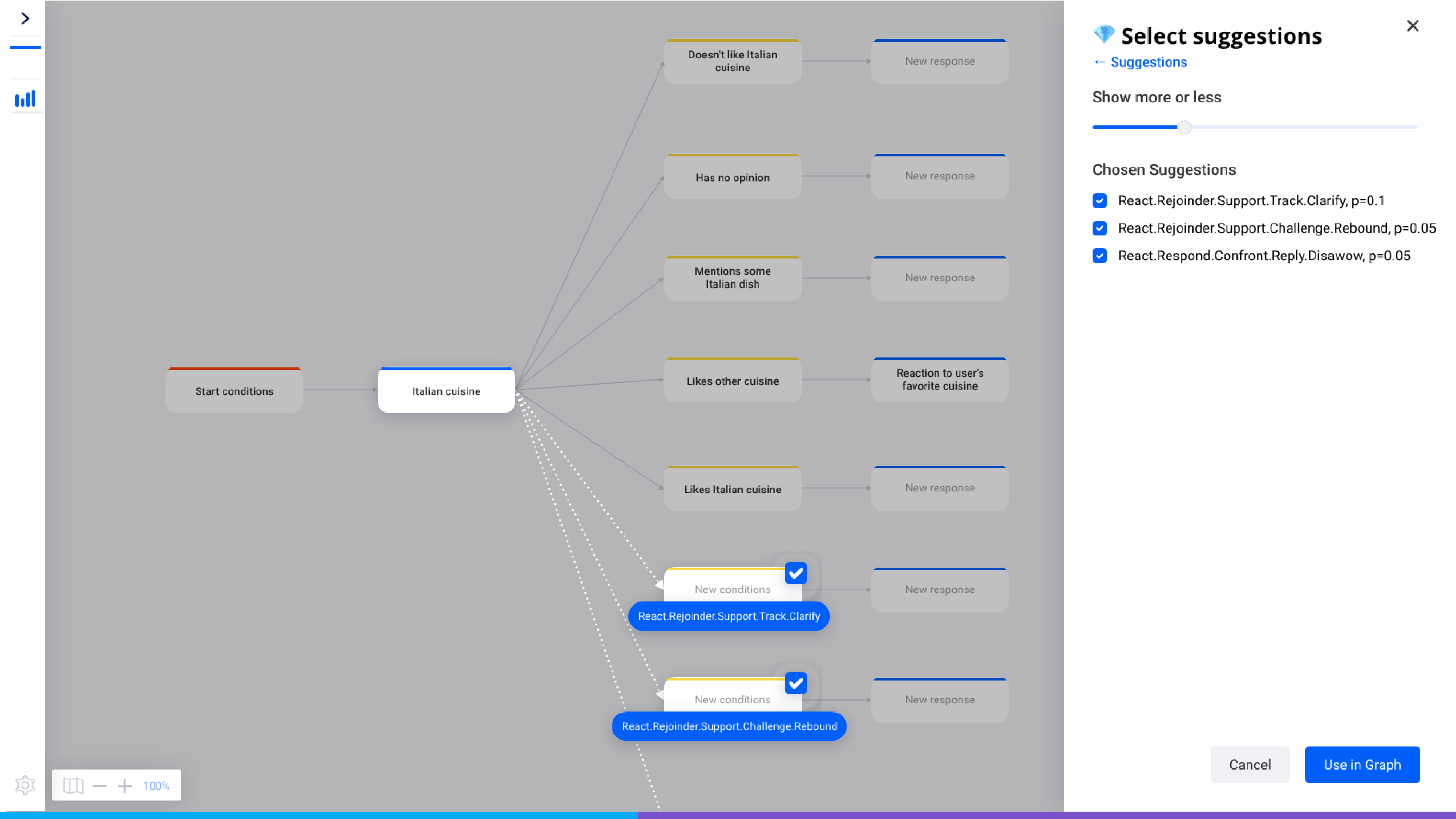
Generate from Classifiers

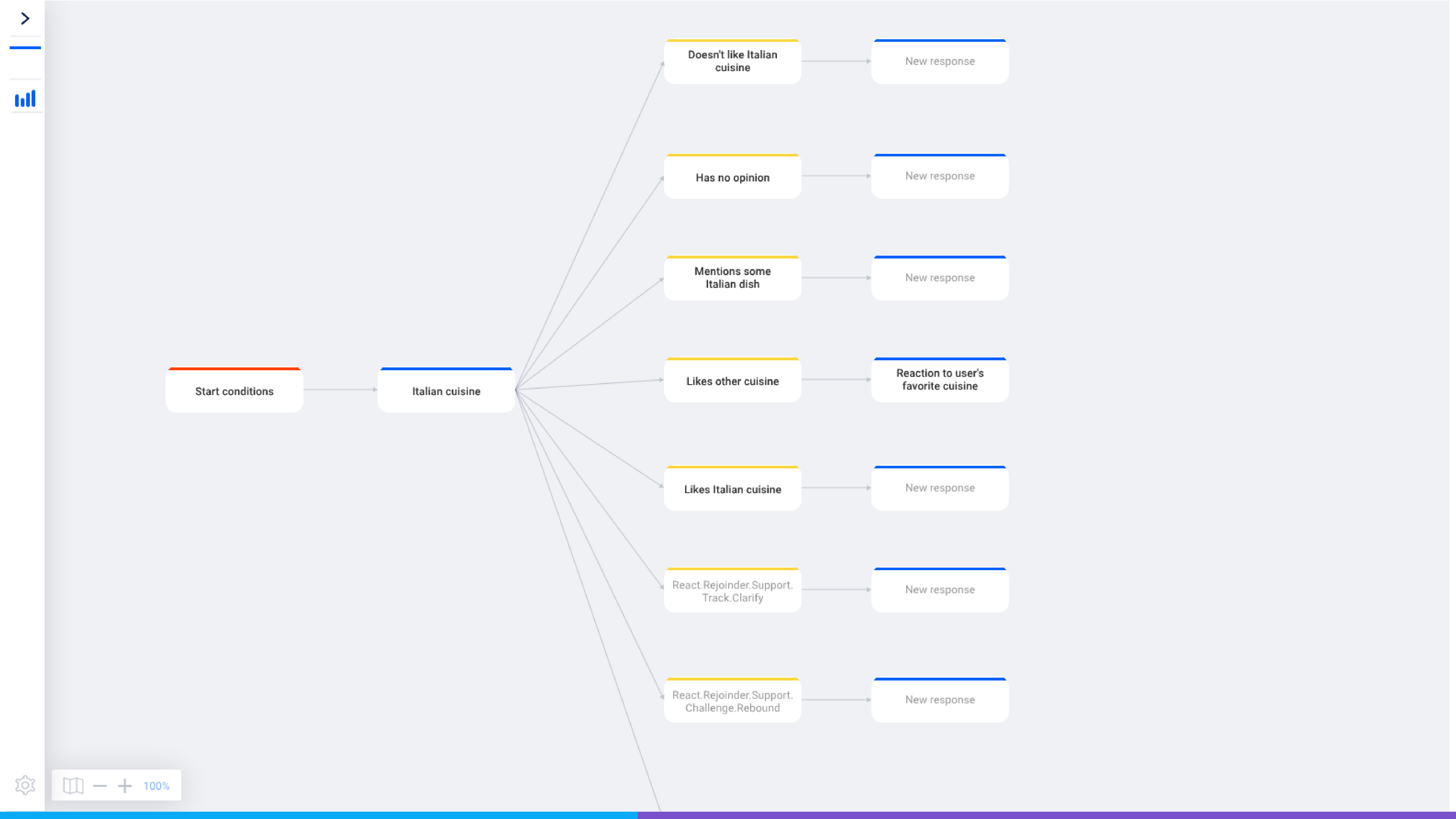
Cancel Save

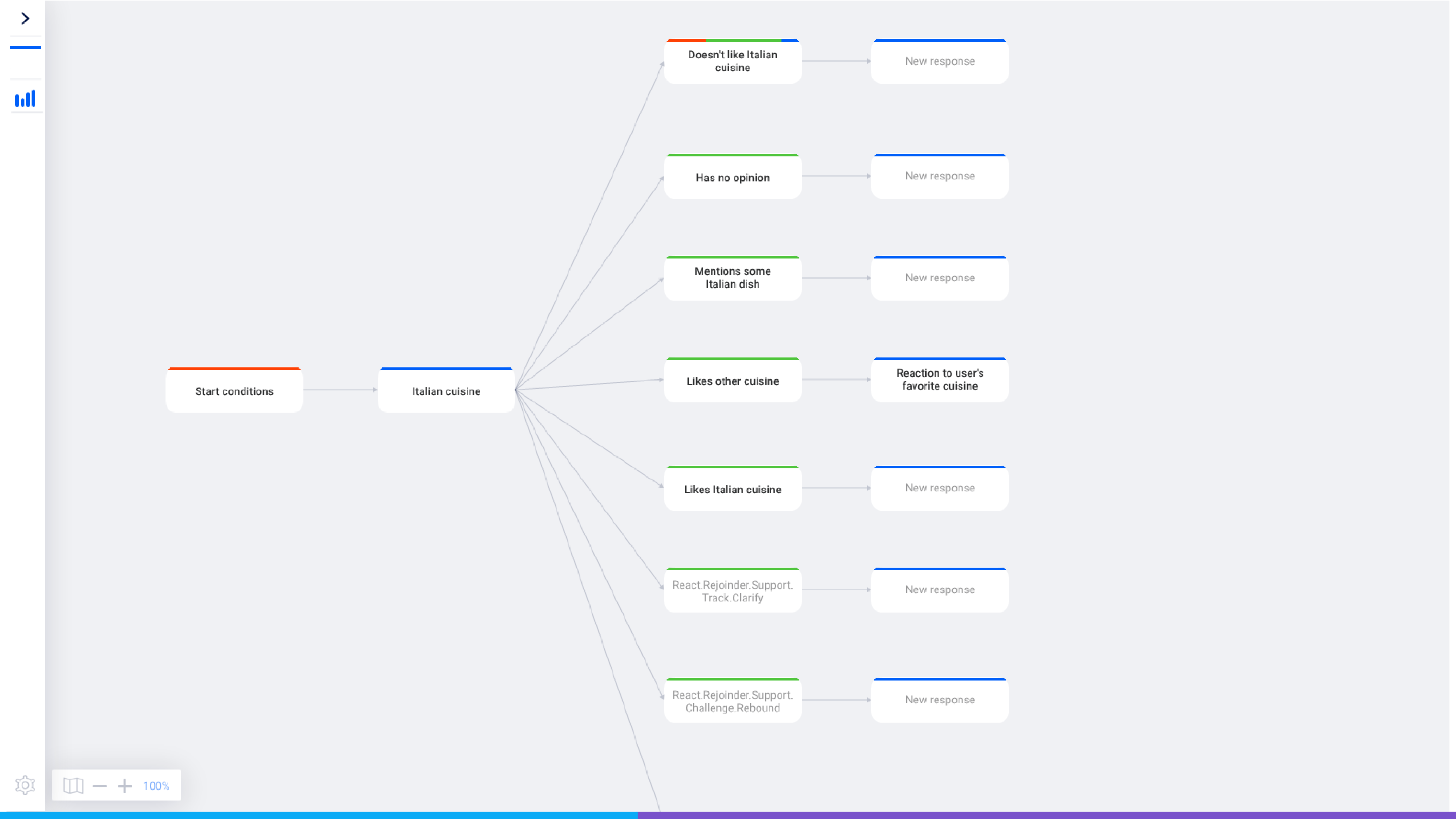

Start conditions
Italian cuisine
Doesn't like Italian cuisine
Has no opinion
Mentions some Italian dish
Likes other cuisine
Likes Italian cuisine
New response
New response
New response
Reaction to user's favorite cuisine
New response
100%
Response
Title
Reaction to user's favorite cuisine
What should the bot say?
Manual
Use Response Generators
Response
Oh, {cuisine}! I just adore it!
Comment
Enter comment (optional)
You can add information about bot's response to use suggestion mode.
Response Parameters
MIDAS is Opinion
sf is React.Respond.Support.Register
sentiment is positive
+ add parameters
Cancel
Save


Doesn't like Italian cuisine
New response
Has no opinion
New response
Start conditions
Italian cuisine
Show suggestions
Edit
Add next conditions
Delete
New response
Reaction to user's favorite cuisine
Likes Italian cuisine
New response
100%

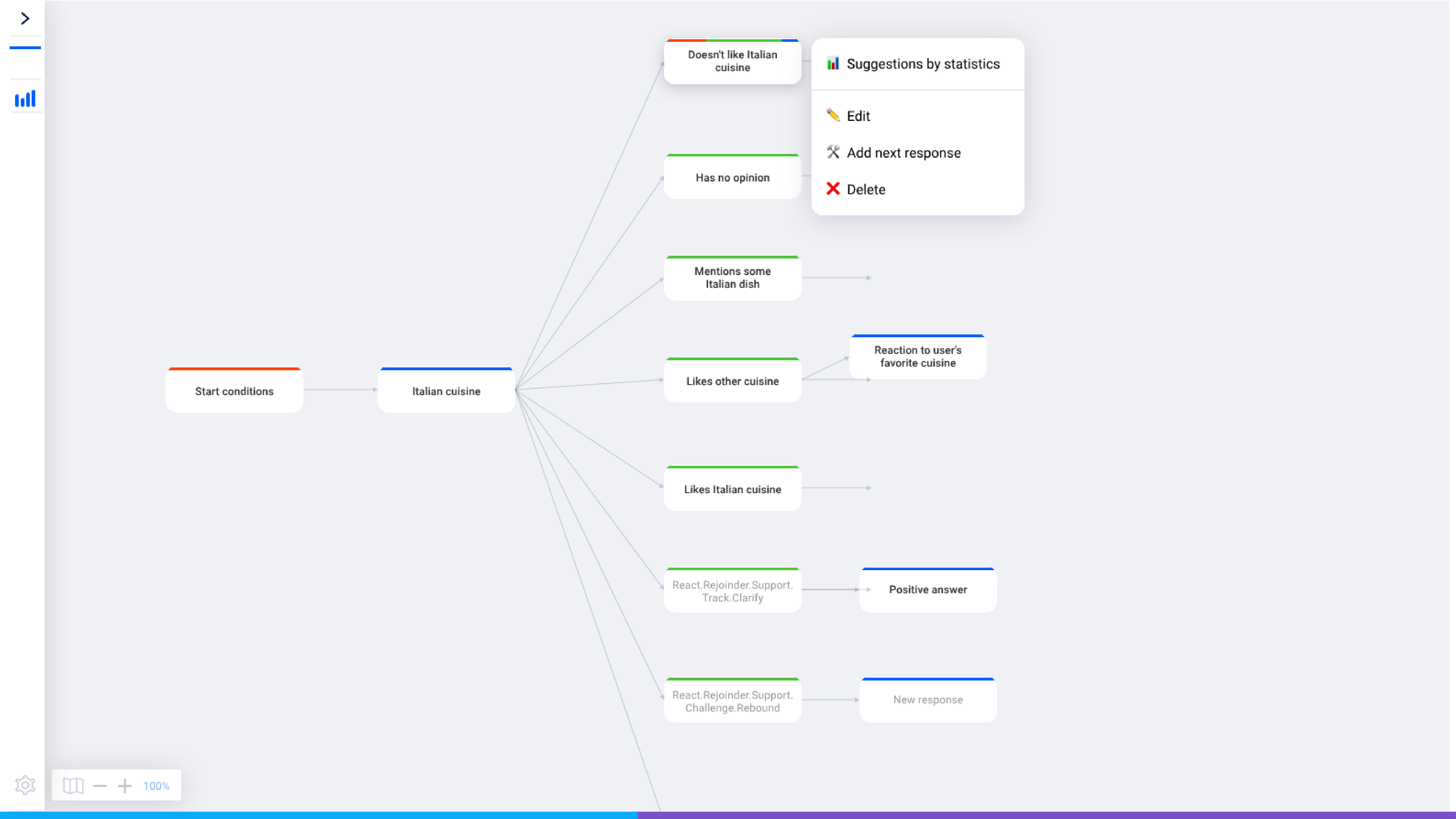

Suggestions
Which suggestions do you want to use?
Discourse Moves Recommendations
Generate Suggestions
Start conditions
Italian cuisine
Doesn't like Italian cuisine
Has no opinion
Mentions some Italian dish
Likes other cuisine
Likes Italian cuisine
New response
New response
New response
Reaction to user's favorite cuisine
New response
100%


Select suggestions
← Suggestions
Show more or less
Chosen Suggestions
React.Rejoinder.Support.Track.Clarify, p=0.1
React.Rejoinder.Support.Challenge.Rebound, p=0.05
React.Respond.Confront.Reply.Disawow, p=0.05
Cancel
Use in Graph
Start conditions
Italian cuisine
Doesn't like Italian cuisine
Has no opinion
Mentions some Italian dish
Likes other cuisine
Likes Italian cuisine
New conditions
React.Rejoinder.Support.Track.Clarify
New conditions
React.Rejoinder.Support.Challenge.Rebound
New response
New response
New response
Reaction to user's favorite cuisine
New response
New response
New response
100%

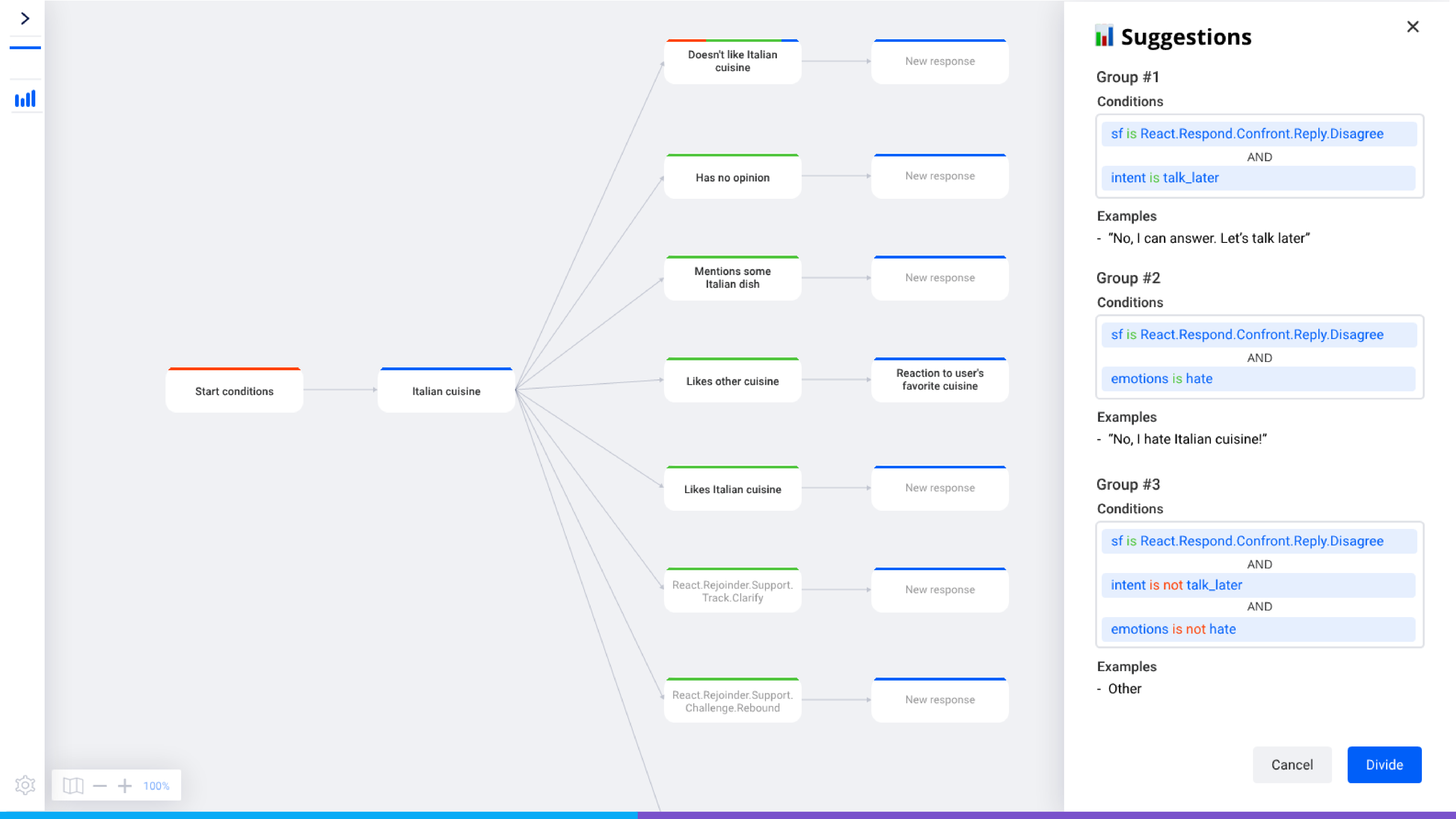

Start conditions
Italian cuisine
Doesn't like Italian cuisine
Has no opinion
Mentions some Italian dish
Likes other cuisine
Likes Italian cuisine
React.Rejoinder.Support.Track.Clarify
React.Rejoinder.Support.Challenge.Rebound
New response
New response
New response
Reaction to user's favorite cuisine
New response
New response
New response
100%



Start conditions
Italian cuisine
Doesn't like Italian cuisine
New response
Has no opinion
New response
Mentions some Italian dish
New response
Likes other cuisine
Reaction to user's favorite cuisine
Likes Italian cuisine
New response
React.Rejoinder.Support.Track.Clarify
New response
React.Rejoinder.Support.Challenge.Rebound
New response
100%

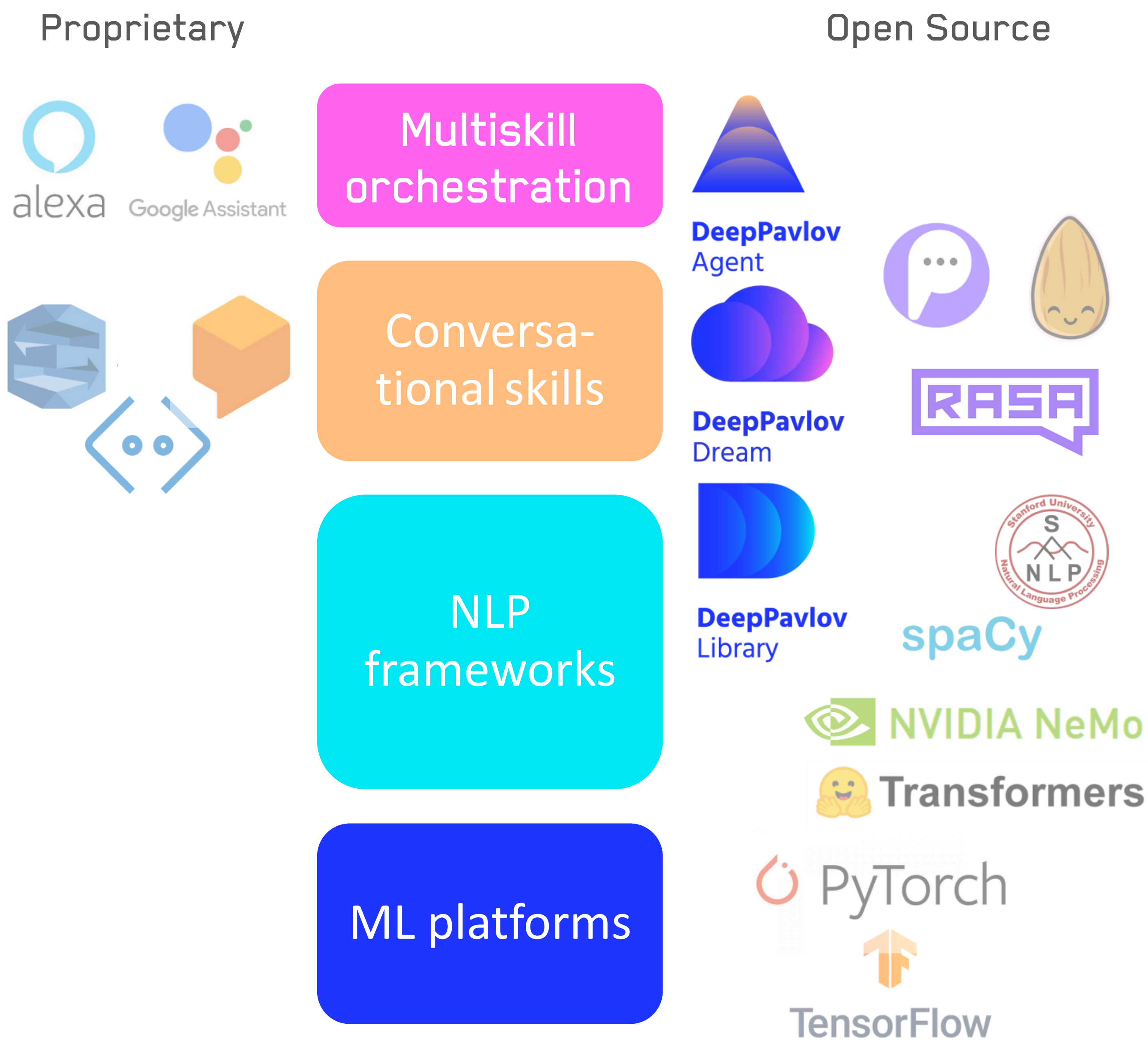


Start conditions
Italian cuisine
Doesn't like Italian cuisine
Has no opinion
Mentions some Italian dish
Likes other cuisine
Likes Italian cuisine
React.Rejoinder.Support.Track.Clarify
React.Rejoinder.Support.Challenge.Rebound
Reaction to user's favorite cuisine
Positive answer
New response
Suggestions by statistics
Edit
Add next response
Delete
100%


Start conditions
Italian cuisine
Doesn't like Italian cuisine
Has no opinion
Mentions some Italian dish
Likes other cuisine
Likes Italian cuisine
React.Rejoinder.Support.Track.Clarify
React.Rejoinder.Support.Challenge.Rebound
New response
New response
New response
Reaction to user's favorite cuisine
New response
New response
New response
100%
Suggestions
Group #1
Conditions
sf is React.Respond.Confront.Reply.Disagree
AND
intent is talk_later
Examples
- "No, I can answer. Let's talk later"
Group #2
Conditions
sf is React.Respond.Confront.Reply.Disagree
AND
emotions is hate
Examples
- "No, I hate Italian cuisine!"
Group #3
Conditions
sf is React.Respond.Confront.Reply.Disagree
AND
intent is not talk_later
AND
emotions is not hate
Examples
- Other
Cancel
Divide

CONVERSATIONAL AI DEEPPAVLOV STACK

- Multiskill Orchestration
  - **DeepPavlov Agent** is an engine for conversational skill deployment and orchestration
- Conversational Skills
  - **DeepPavlov Dream** - collection of pre-build conversational skills and a default AI Assistant distribution package
  - **DeepPavlov DFF** - framework for building scenario-driven chatbots & **Dream** skills
- NLP Frameworks
  - **DeepPavlov Library** - pretrained models and both Python & declarative approaches for building NLP pipelines
- ML Platforms
  - TensorFlow and PyTorch as backends



DeepPavlov Library

DeepPavlov Agent

DeepPavlov Dream

DEEPPAVLOV CONVERSATIONAL AI STACK

What's next?

DeepPavlov Library

DeepPavlov Agent

DeepPavlov Dream

DEEPPAVLOV CONVERSATIONAL AI STACK

Extending our product family!

DeepPavlov Library

DeepPavlov Agent

DeepPavlov Annotators

DeepPavlov DFF

DeepPavlov DFF SDK

DeepPavlov DFF Designer

DeepPavlov Dream

DEEPPAVLOV CONVERSATIONAL AI STACK

Welcoming our new components: Annotators in DockerHub, DFF, DFF SDK, and DeepPavlov DFF Designer!

THANKS TO THE TEAM!

Denis Kuznetsov

DFF

Dmitry Evseev

DD-IDDE SDK

Lida Ostyakova

Speech Functions

Bálint Magyar

VS Code Extension

Dmitry Babadeev

VS Code Extension

Ksenia Petyxova

Skill Writer

Nika Smilga

Skill Writer

Oleg Serikov

VS Code Extension

Maria Molchanova

Product Manager

Daniel Kornev

Deputy Head of Lab

Mikhail Burtsev

Head of Lab

SPECIAL THANKS:

Yuri Kuratov

Senior Researcher

WELCOME TO DEEPPAVLOV!

Read us: medium.com/deeppavlov
Twitter/TG: @DeepPavlov

Talk to us: forum.deeppavlov.ai
TG: @DeepPavlovDreamDiscussions

DD–IDDE: bit.ly/dd-idde or look up DD-IDDE in VS Code

DD–IDDE SDK: bit.ly/dd-idde-sdk

Talk to us about DD–IDDE: bit.ly/dd-idde-qa or use QR code:

Come join us!